Program	Bachelor of Technology (BTech)	**Semester - 4**
Type of Course	Professional Electives	
Prerequisite	Basic coding, Web Development, JavaScript, and Object Oriented Programming	
Course Objective	The .NET has become a platform of choice for the development of web-based data-driven pages It is a popular platform for the development of robust enterprise and web-based applications. In this course, students will be able to use the ASP.NET platform for developing web-based applications with database support. This course aims to enable students to develop dynamic and data-driven web applications utilizing the power of .NET Technology.	

Teaching Scheme (Contact Hours)				**Examination Scheme**				
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory Marks**		**Practical Marks**		**Total Marks**
				SEE (T)	**CIA (T)**	**SEE (P)**	**CIA (P)**	
3	0	4	5	40	30	20	10	100

***SEE** - Semester End Examination, **CIA** - Continuous Internal Assessment (It consists of Assignments/Seminars/Presentations/MCQ Tests, etc.)*

Course Content

T - Teaching Hours | **W** - Weightage

Sr.	**Topics**	**T**	**W**
1	**C# Fundamentals**	**10**	**20**
	Overview, Namespace, Variables, Data Types, Operators, Strings, Conditions & Looping, Collection Classes, Exception Handling, Classes/Objects, Constructors, Access Modifier, Properties, Inheritance, Interface		
2	**Introduction to ASP.NET Core**	**8**	**20**
	Overview, ASP.NET Core Features, Advantages, MVC Architecture Pattern, Environment Setup, Project Layout, ASP.Net Core Life Cycle, Request Response Pipeline & Middleware **Areas, Controllers & Action Methods:** Understanding Areas, Adding Areas, Defining Area Routes, Linking Between Areas, Controllers Overview, Action Methods and IActionResult object, Understanding Action Selectors, Action Filters, Non Action Methods		
3	**Views & Helpers**	**10**	**20**
	Introducing Razor View, Advantages of Razor View, Razor Syntax, Types of Views, Partial Views, Layout Pages, Helpers Overview, Html Helpers, Built-In Html Helpers, URL helpers, Tag Helpers, Custom Tag Helpers		
4	**Data Annotations, Validations, Model Binding & Data Passing**	**9**	**20**
	Validation Overview, Validations with Data Annotation, Client Side and Server-Side Validation, Html Form behavior, Default Model Binder, IFormCollection Model Binding, Bind Attribute, Data Passing Techniques		
5	**Routing, State Management & ADO.NET**	**8**	**20**
	Routing Overview, Custom Routes, Attribute Routing, Routing Constraints, ADO.NET Architecture, .NET Framework Data Providers, Connection & Command Object, DataReader Object, Query Strings, Session		
	Total	**45**	**100**

Suggested Distribution Of Theory Marks Using Bloom's Taxonomy

Level	Remembrance	Understanding	Application	Analyze	Evaluate	Create
Weightage	30	40	30	0	0	0

NOTE : This specification table shall be treated as a general guideline for the students and the teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcomes

At the end of this course, students will be able to:	
CO1	**discuss** the basic concepts of .NET Framework and fundamentals of C# programming language
CO2	**classify** the MVC Pattern, Controllers & Action Methods
CO3	**perform** views with different layouts with tag helpers
CO4	**implement** data annotations for model validations
CO5	**apply** ADO.Net objects/classes to communicate with database

Reference Books

1.	**Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming** By Andrew Troelsen, Phil Japikse | Apress
2.	**Programming Asp.Net Core** By Dino Esposito | Microsoft Press
3.	**Asp.Net Core Application Development :: Building An Application In Four Sprints** By David Paquette, Simon James Chambers | Microsoft Press
4.	**ASP.NET Core Razor Pages in Action** By Mike Brind | Manning

List of Practical

1. **To implement concept of Class and Object, Constructors, Inheritance**

1.(A) Write a program to create a class named Candidate with ID, Name, Age, Weight and Height as data members & also create a member functions like GetCandidateDetails() and DisplayCandidateDetails().
2.(A) Write a program to create a class Staff having data members as Name, Department, Designation, Experience & Salary. Accept this data for 5 different staffs and display only names & salary of those staff who are HOD.
3.(A) Write a pogram to Create a class Bank_Account with Account_No, Email, User_Name, Account_Type and Account_Balance as data members. Also create a Member function GetAccountDetails() & DisplayAccountDetails().
4.(A) Write a program with following specifications:
Class Name: Student
Data Members: Enrollment_No, Student_Name, Semester, CPI and SPI
Get Students Details using constructor and DisplayStudentDetails() using member function.
5.(A) Write a program to calculate area of a Rectangle using constructor.
6.(A) Write a program for implementing single inheritance which creates one class Account_Details for getting account information and another class
Interest for calculating and displaying total interest from the data inserted
from account details.
7.(B) Write a program to Define a class Salary which will contain member variable Basic, TA, DA, HRA. Write a program using Constructor with default values for DA and HRA and calculate the salary of employee.
8.(B) Write a program to Define a class Distance have data members dist1, dist2, dist3. Initialize the two data members using constructor and store their addition in third data member using function and display addition.
9.(C) Create a class Furniture with material ,price as data members. Create another class Table with Height , surface_area as data members. Write a program to implement single inheritance.
10.(C) Program to implement the following multiple inheritance using interface.
Interface: Gross
Method- Gross_sal()
Class : Salary
Data Members - HRA, TA,DA
Methods - Disp_sal() Class : Employee
Data Members - Name
Methods - basic_sal()

2. Implement concept of Exception Handling, Interface, Abstraction, String Functions

1.(A) Write a program to Create a divide by zero exception and handle it.
2.(A) Write a program that reads 5 numbers from user. Demonstrate concept of IndexOutOfRange Exception.
3.(A) Write a program to create an abstract class Sum having abstract methods SumOfTwo(int a, int b) and SumOfThree(int a, int b,int c). Create another class Calculate which extends the abstract class and implements the abstract methods.
4.(A) Write a program to create interface Calculate. In this interface we have two member functions Addition() and Subtraction(). Implements this interface in another class named Result.
5.(A) Write program showing use of common methods of String class.
6.(B) Write a program to Replace lower case characters to upper case and Vice-versa.
7.(B) Write a program to create interface named Shape. In this interface, we have three methods Circle(), Triangle() and Square() which calculates area of Circle, Triangle and Square respectively. Implement Shape interface.
8.(B) Write a program to accept a number from the user and throw an exception if the number is not an even number.
9.(C) Write a program to find the longest word in a string.
10.(C) Write a program to change the case of entered character.

3. Write program using Method Overloading, Method Overriding & Delegate

1.(A) Write a program using method overloading by changing datatype of arguments to perform addition of two integer numbers and two float numbers.
2.(A) Write a program using method overloading by changing number of arguments to calculate area of square and rectangle.
3.(A) Create a class named RBI with calculateInterest() method. Create another classes HDFC, SBI, ICICI which overrides calculateInterest() method.
4.(B) Write a program to return the factorial from the method using delegate.
5.(B) Create a class Hospital with HospitalDetails() method. Create another classes Apollo, Wockhardt, Gokul_Superspeciality which overrides HospitalDetails() method.
6.(C) Write a programs to Find Area of Square, Rectangle and Circle using Method Overloading.
7.(C) Write a program to create a delegate called TrafficDel and a class called TrafficSignal with the following delegate methods.

```
Public static void Yellow()
{
    Console.WriteLine(“Yellow Light Signal To Get Ready”);
}
Public static void Green()
{
    Console.WriteLine(“Green Light Signal To Go”);
}
Public static void Red()
{
   Console.WriteLine(“Red Light Signal To Stop”);
}
```

8.(C) Write a program to create a delegate calculator to demonstrate the example of generic delegate.

4. To demonstrate concept of Collection Classes

1.(A) Create an ArrayList for StudentName and perform following operations:
  a. Add() - To Add new student in list
  b. Remove() - To Remove Student with specified index
  c. RemoveRange() - To Remove student with specified range.
  d. Clear() - To clear all the student from the list
2.(A) Create a List for StudentName and perform following operations:
  a. Add() - To Add new student in list
  b. Remove() - To Remove Student with specified index
  c. RemoveRange() - To Remove student with specified range.

	d. Clear() - To clear all the student from the list 3.(B) Create a Stack which takes integer values and perform following operations: a. Push() - To Add new item in stack b. Pop() - To Remove item from the stack c. Peek() – To Return the top item from the stack. d. Contains() - To Checks whether an item exists in the stack or not. e. Clear() - To clear items from stack 4.(C) Create a Queue which takes integer values and perform following operations: a. Enqueue() - Adds an item into the queue. b. Dequeue() - Returns an item from the beginning of the queue and removes it from the queue. c. Peek() - Returns an first item from the queue without removing it. d. Contains() - Checks whether an item is in the queue or not e. Clear() - Removes all the items from the queue
5.	**To setup project environment and create a simple calculator** 1. (A) Introduction to IDE, how to create project of .net core, how to add controllers, action methods and views. 2. (A) How to add NuGet package references? 3. (B) Project Creation & View Creations (Home, AboutUs,ContactUs) 4. (C) Create Simple Calculator.
6.	**Demonstrate concept of Razor Syntax by displaying student details** 1.(A) Prepare a page which displays student details and his/her semester wise SPI in table format. 2.(A) Prepare semester wise SPI table data in controller file and store it in ViewBag. Display the data to view page using foreach loop. 3.(C) Print Table of 5 using Razor Syntax.
7.	**Implementation of concept of View, Controllers and Action Methods** 1.(A) Prepare employee page which displays employee details in table format. Create employee model class for it and use List collection class object to pass data from controller to view. 2.(A) Add delete functionality in table page. 3.(A) Add functionality to insert a record. 4.(A) Add functionality to update a record. 5.(B) Update on Specific Fields. 6.(C) Delete Many Using Multiple.
8.	**Implement concept of Routing** 1.(A) Create a project and add Home, About, Contact Us views. 2.(A) Appropriate routing between these pages. Use bootstrap for better design. (use conventional routing) 3.(B) Add attribute routing in above practical along with optional parameters and apply ignore route templates functionality. 4.(C) Routing in Personal Website (CV)
9.	**Create Database and prepare stored procedures for Select command** 1.(A) Create Database : StudentMaster also Create all tables SelectAll and SelectByPK stored procedures 2.(B) Create Custom Parameter Procedures 3.(C) Create Procedures with Null & Compulsory Parameters
10.	**Prepare stored procedure for Insert, Update and Delete command** 1.(A) Create all tables Insert, Update and Delete stored procedures 2.(B) Create Custom Parameter Procedures 3.(C) Create Procedures with Null & Compulsory Parameters
11.	**Demonstrate concept of layout page** 1.(A) Single page bootstrap theme conversion using View pages and Layout page. 2.(B) Also convert admin theme for the project with required pages. [Use Nice Admin Template] 3.(C) Custom Template Theme Conversion

12.	**Hands on practice to standard HTML helpers and strongly typed HTML helpers by creating student registration form** 1.(A) Student registration form using Standard html helpers.(StudentName, Branch, Semester, Birthdate, Mobile, Email, Address, City, Hobbies, Gender) 2.(A) Student registration form using Strongly typed html helpers.(StudentName, Branch, Semester, Birthdate, Mobile, Email, Address, City, Hobbies, Gender) 3.(B) Employee Add Form using Both Helpers. 4.(C) Admission Form using Strongly typed html helpers.
13.	**Perform of file upload operation** 1.(A) Perform file upload operation and display the uploaded file. 2.(B) Perform multiple file upload operation and display it. 3.(C) Perform Drag & Drop Multiple file upload.
14.	**Create a project and implement area** 1.(A) Create a new asp.net core project with MVC Template. 2.(B) Create appropriate MVC Areas for Country, State, City, Branch, and Student 3.(C) Create Employee Area & CRUD operations with Routing
15.	**Create design for view pages** 1.(A) Design List Page & Add/Edit Pages. [For LOC_Country, LOC_State, LOC_City] 2.(B) Design for Student Registration Form CRUD using theme
16.	**Implement attribute routing between view pages** 1.(A) Apply Attribute Routing in whole Project 2.(B) Apply Conventional Routing
17.	**Database connectivity and Implementation of read operation** (A) Create Database connectivity and Display data (All Records) for LOC_CountryList.cshtml, LOC_StateList.cshtml, LOC_CityList.cshtml view pages
18.	**Create a model class to implement concept of data annotation** (A) Implement data annotation on all the model classes.
19.	**Implementation of delete functionality** (A) Implement Delete functionality for LOC_Country, LOC_State, LOC_City with prompt Are you sure you want to delete record?
20.	**Perform server side validation** (A) Apply server side validations with proper message.
21.	**Implementation of insert functionality** (A) Implement Insert functionality for LOC_Country, LOC_State, LOC_City view pages with required validations
22.	**Implementation of update functionality** (A) Implement Update functionality for LOC_Country, LOC_State, LOC_City view pages.
23.	**Demonstrate concept of session** (A) Implement Session functionality in your project. [Add Login and Sign Up]
24.	**Implementation of Search functionality** (A) Implement Search functionality for all the list pages.
25.	**Implement functionality to change user profile** (A) Implement functionality by which user can change their profile.
26.	**Implementation of cascade dropdown functionality** (A) Implement functionality to fill state wise city using cascade dropdown functionality.
27.	**Sweet alert implementation**

	(A) Implement Sweet alert when user want to delete any record.
28.	**Dashboard design and Export excel functionality** 1.(A) Display Different counts & Statistics using Temporary Table and Temporary variables. 2.(A) Add a button by which user can export table data to excel.
29.	**Implement functionality to Delete many records using single click** (A) Add checkboxes in table as column. User can delete multiple selected records.
30.	**Demonstrate concept of URL encryption and decryption** (A) Perform URL Encryption and Decryption using standard encryption decryption algorithm.